Measuring the Likelihood Property of Scoring Functions in General Retrieval Models

Richard Bache,
Dept. Computer and Information Science,
University of Strathclyde,
16, Richmond Street,
Glasgow, G1 1QX,
Scotland.
Tel: +44 141 548 3081
Fax: +44 141 548 4523
e-mail: richard.bache@cis.strath.ac.uk

Mark Ballie,
Dept. Computer and Information Science,
University of Strathclyde,
16, Richmond Street,
Glasgow, G1 1QX.
Scotland
Tel: +44 141 548 3705
Fax: +44 141 548 4523
e-mail: mark.ballie@cis.strath.ac.uk

Fabio Crestani (contact author),
Faculty of Informatics,
University of Lugano,
Via G. Buffi, 13,
Ch-6900 Lugano,
Switzerland.
Tel: +41 58 666 4657
Fax: +41 58 666 4536
e-mail: fabio.crestani@unisi.uk

Abstract: Although retrieval systems based on probabilistic models will rank the objects (e.g. documents) being retrieved according to the probability of some matching criterion (e.g. relevance) they rarely yield an actual probability and the scoring function is interpreted to be purely ordinal within a given retrieval task. In this paper it is shown that some scoring functions possess the likelihood property, which means that the scoring function indicates the likelihood of matching when compared to other retrieval tasks which is potentially more useful than pure tanking although it cannot be interpreted as an actual probability. This property can be detected by using two modified effectiveness measure, entire precision and entire recall. Empirical evidence is offered to show the existence of this property both for traditional document retrieval and for analysis of crime data where suspects of an unsolved crime are ranked according to probability of culpability.

1. Introduction

Retrieval systems in general and Information Retrieval (IR) systems in particular typically produce a ranked list of objects (e.g. documents) according to some search criterion (e.g. relevance). Where this criterion is a dichotomous attribute it is meaningful to talk of the probability that some object satisfies it. However, as we see below an estimate of probability is often not possible. Instead we consider a weaker but still useful property of *likelihood* which is above and beyond mere ranking of objects from some retrieval task. It allows us to consider that an object is more or less likely to match some criterion irrespective of its position in the list. In this paper we firstly define the likelihood property, secondly propose a way of measuring it and thirdly provide three cases where a scoring function can be shown to exhibit the likelihood property: one when analysing TREC collections and two which are based on the analysis of crime data.

In the traditional IR domain there will be situations where the scoring function not only orders the documents in a collection according to an estimated measure of relevance to a query but also implies a notion of relevance likelihood. Then, in addition to the standard retrieval task, it has the following uses:

- The user can be guided how far down the list to search for relevant documents indicating when the likelihood of a document being relevant to the query is higher or lower than for previous queries;
- It can be used for pseudo-relevance feedback. Relevance feedback is where documents already retrieved and known to be relevant are used to modify the query in order to perform a new scoring of the collection. Pseudo-relevance feedback assumes that the top ranked documents are relevant and uses them without an independent relevance judgement. For this latter case only documents with a likelihood measure over a given threshold need be chosen;
- By examining the scores for the top-ranked documents it is possible to gauge query difficulty providing an alternative to the approach proposed by Carmel et al. (2006).

We do not propose that all scoring functions exhibit this property – most do not. But Bache et al. (2007b) do propose such a scoring function based on Language Modelling

(Ponte & Croft, 1998). We propose therefore a way of measuring whether this likelihood property is present for a given scoring function over standard sets of data.

Many of the concepts applied in IR also prove useful in the analysis of crime data and the likelihood property proves useful in this domain since it gives an indication of the degree of trustworthiness when models seek to identify the culprit for an unsolved crime. This research arose out of the EPSRC (UK Government) funded project iMOV – a multi-disciplinary collaboration between computer scientists and investigative psychologists. The motivation for the work presented here arose out of the problem of prioritisation of suspects which we now define. Given an unsolved crime and a list of suspects who are known to have committed this type of crime before, we rank the suspects according to how likely each suspect is to be the actual culprit. We assume that there is some information about each suspect's past criminal history and about the unsolved crime; we are thus making a closed world assumption in that all possible suspects are known. Linking a suspect with a crime can be done on the basis of the observed actions of that offender where we have textual information describing the unsolved crimes and past solved crimes. Linking can also be performed where we have the physical location of each crime as a coordinate and not text at all. Thus some scoring function attempts to order the suspects according to the strength of linkage between the past crimes and the unsolved one. For example, in the case of a burglary, if only one past offender is known to enter a house by forcing a door with a crowbar and this behaviour is repeated in the unsolved crime, this links the suspect to this crime. Also, if a burglar is known to operate in the area around the unsolved crime, again it links the suspect to the crime; this latter case is using coordinates rather than free text. The top ranked suspect is, by definition, the one considered most likely to be the culprit. If the scoring function possesses the likelihood property then the magnitude of the scoring value also indicates how much more likely this suspect will be over the other suspects lower down the list.

The rest of the paper is organised as follows: Section 2 explains a general retrieval system based on Fuhr's Conceptual Model (1992) and then defines the likelihood property. In Section 3 we argue that likelihood can be assessed using the proposed measures of entire precision and entire recall. Section 4 applies these measures to two scoring functions and 3 TREC collections. Section 5 considers suspect prioritisation where crimes are linked by free text descriptions of the offences. In Section 6, we look at suspect prioritisation where crimes are linked by distance. Section 7 provides some conclusions. Note that Sections 5 summarises results published elsewhere, whereas Sections 4 and 6 provide results for the first time. However this is the first time that all three applications have been presented together as a general theory.

2. General Retrieval Systems

In traditional IR we have a collection of *documents* and a set of *queries* and seek to determine if documents are *relevant* to a specific query (or more accurately the information need which led to the query's formulation). The *scoring function* (also called a *ranking function* or *matching function*) is an attempt to give a numerical value to each document according to its relevance; these score are usually used to rank the documents in order of relevance. Relevance is, in a sense, outside the system and actual judgements of relevance are made by humans. One important way of evaluating an IR system is by measuring its *effectiveness* – that is the degree to which the scoring function agrees with the human relevance judgements. However, since the likelihood property discussed here is applicable to areas beyond this domain, we will need a more general terminology.

2.1 Beyond Queries, Documents and Relevance

Instead of *documents* and *queries*, we shall speak of *objects* and *keys* since this will allow us to deal with geographical information where there is no actual text. Fuhr (1992) proposes a conceptual model for IR and this is clarified further by Crestani et al. (1998). We now generalise this model to deal with the more general situation described here but keep many aspects of Fuhr's notation.

We start with assuming collection (set) of *objects* $\underline{O}$ and a set of *keys* $\underline{K}$. In the case of IR these are respectively the set of documents and information needs that users may have. There is some *matching criterion* which links each object with each key which we can interpret as relevance in the traditional IR domain. There are functions α_O and α_K which map the objects and keys to *representations* thereof, *O* and *K*. In IR, $o \in O$ is the content of the document or of some surrogate (such as a review or abstract); $k \in K$ is the text of the query composed by the user. There are two further functions, β_O and β_K, which map the representations to *descriptions O'* and *K'*. These can be interpreted as the indexes of the text which could be derived automatically and reduce free text to a vector of terms optionally with stemming and stopword removal. However the model is consistent with manual indexing too. There is a *scoring function* which gives a real-valued score for each object and key. $r : O' \times K' \to \Re$. Figure 1 shows the relationship between these entities.

$$\begin{tikzcd}
 & \underline{K} \arrow[r, "\alpha_K"] \arrow[dl] & K \arrow[r, "\beta_K"] & K' \arrow[dr] & \\
M & & & & \Re \\
 & \underline{O} \arrow[r, "\alpha_O"] \arrow[ul] & O \arrow[r, "\beta_O"] & O' \arrow[ur] &
\end{tikzcd}$$

Figure 1: A Generalisation of Fuhr's Conceptual Model

We later see that this model can be extended for crime data. Here, $\underline{K}$ is the set of unsolved crimes and $\underline{O}$ is the set of suspects who have previously committed at least one crime of this type. The matching criterion *M* is culpability – that a suspect committed that crime (possibly with accomplices). The full interpretation of the model is detailed in sections 5 and 6, but it is sufficient to say here that both for the text-based and geographical representation of crimes we are able to define a scoring function.

2.2 Probability of Matching

Throughout this paper, we will consider the special case where the matching criterion is dichotomous; either an object and key match or do not $M = \{m, \overline{m}\}$. This is common assumption used in IR evaluations. For example, the majority of TREC evaluations (Voorhees & Harman, 2005) adopt this view of the world. This is also a reasonable assumption for suspect prioritisation since a suspect is either innocent or guilty.

We now address what the scoring function actually means. No scoring function can realistically achieve a perfect partition the set of objects into matching and non-matching subsets. As Crestani et al. (1998) point out there is a fundamental difference between scoring functions which seek to measure similarity between the object and key (e.g. vector space models) and a scoring function derived from the probability of a document being relevant to a query. The likelihood property is only meaningful in this latter case. Although both similarity and probabilistic scoring functions are used for traditional IR, the models applied to crime data proposed here are necessarily Bayesian and thus assume a probabilistic interpretation.

2.3 Definition of Likelihood Property

According to Robertson's Probability Ranking Principle (1997), a scoring function can only rank documents according to the probability of relevance when the following condition is true.

$$r(o_1', k') > r(o_2', k') \Rightarrow P(m \mid o_1', k') > P_m(m \mid o_2', k') \qquad (1)$$

where $o_1', o_2' \in O'$ are descriptions of objects;

$k' \in K'$ is a description of a key and

$P(m \mid o', k')$ is the probability of object, *o'*, matching key, *k'*.

This means that two objects will be ranked by the probability that each matches a given key. In the context of information retrieval, where objects are documents and keys are queries, document 1 would be ranked over document 2 for a given query implies that the probability of document 1 being relevant to that query is higher than the probability that document 2 is relevant to the same query. This is the weakest pnecessary for a function derived from the notion of probability of matching to qualify as a scoring function at all.

The strongest condition we can impose is that the scoring function actually yields the probability of matching.

$$r(o', k') = P(m \mid o', k') \qquad (2)$$

This is the best situation we could achieve given that we are basing the scoring function on descriptions (or indices) of the document and key. As we see below, this is very difficult to achieve.

The likelihood property is defined to exist when the following condition holds.

$$r(o_1', k_1') > r(o_2', k_2') \Rightarrow P(m \mid o_1', k_1') > P(m \mid o_2', k_2') \qquad (3)$$

where $k_1', k_2' \in K'$. This means that two objects will be ranked by the probability that they match two (possibly distinct) keys. So the scoring function provides a measure of likelihood that may be applied for different keys although this measure may fall short of being an actual probability. We note that Condition 2 implies Condition 3 which, in turn, implies Condition 1.

In essence, Condition 3 implies that the score is comparable not only within the objects retrieved for one key but also across keys. In the context of Information Retrieval, if the scoring function adhered to this condition, we would be able to compare the relevance of documents across queries. This would be beneficial for the three proposed uses specified in Section 1. Users could be guided how far down a ranked list it would be useful to browse in terms of relevance i.e. if the relevance score falls below a given value. This threshold would have to be determined by the success of other queries to this particular collection (and possibly in general). As a means of assisting pseudo-relevance feedback, we only select those documents above a certain likelihood score and if the score for the top ranked item is sufficiently low we may decide that feedback is counterproductive. Again this threshold would have to be calibrated from the experience of other queries. For assessing query difficulty, if the relevance scores in a ranked list were below those typically attained for other (successful) queries, this may be an indicator that the query did not retrieve many relevant documents.

In the context of suspect prioritisation (either from text descriptions or geographical locations), Condition 3 implies that the suspects towards the top of the list with a high score, and in particular the prime suspect at the top of the list, are more likely to be culprit. Consider two unsolved crimes A and B where the respective scores given to the prime suspects are 0.3 and 0.9. We would be able to infer that the prime suspect for crime

B was more likely to be the culprit and thus, given limited resources, prioritise the investigation into crime B.

An interesting corollary of condition 3 is

$$r(o',k_1') > r(o',k_2') \Rightarrow P(m \mid o',k_1') > P(m \mid o',k_2') \qquad (4)$$

which means that where two keys are used to retrieve the same object if k_1' scores higher than k_2' it is more likely to match the object. In other words, for a given object we can rank the queries in order of probability of matching.

2.4 Testing for the Presence of the Likelihood Function

It is possible to argue that the likelihood property exists purely from the assumptions used to derive the scoring function. Indeed we would expect a theoretical argument to underpin such a view that the property existed. However, all probabilistic models are based on a set of assumptions which are often idealised. For example, in Language Modelling, it is assumed that the occurrence of terms (i.e. words) is independent. Common sense tells us that some words are more likely to co-occur with other words and so this assumption is plainly false. Nevertheless, despite a mismatch between the assumptions of a model and the more messy reality of the world, such models do yield useful results. Therefore if we accept that the assumptions underlying these models are, at best, approximations to the truth then there would still be an obligation to show that the likelihood property actually existed. An implication of Condition 3 is that the actual value of the scoring function is a better predictor of matching than the ranked value of objects implied by the scoring function for a single key. Here we are comparing the estimate of a probability of matching with the grounded truth which is that a key and object either match or they do not. Therefore any evidence can only increase our belief that the likelihood property exists although in some cases this evidence will be overwhelming. Standard methods for assessing retrieval effectiveness cannot be used to assess if Condition 3 applies. As we discuss below, they only assess the extent to which Condition 1 is true. We therefore propose new measures.

3. Entire Precision and Entire Recall

Precision and recall are standard measures used to gauge the effectiveness of an IR system (van Rijsbergen, 1979). Precision is the number of retrieved objects that match the key as a proportion of the total number of retrieved objects. Recall is the number of retrieved objects that match as a proportion of the number of matching objects. In the simplest situation a scoring function has just two values and thus partitioning the collection into *retrieved* and *not retrieved* is trivial. More often the scoring function will give a range of values and so partitioning can be achieved by setting some cut-off point; objects above the cut-off are deemed to be retrieved. As the threshold is lowered more objects are deemed to be retrieved and recall will rise as precision tends to fall. In this situation various measures derived from precision and recall are used. For example, a common approach is to produce precision/recall graphs. Composite measures such as average precision, which is, loosely speaking, the area under the precision/recall graph are often used. Alternatively, precision is calculated at various cut-off points are calculated and plotted, e.g. precision at 10, 20, 50, 100 etc.

Where the scoring function yields a range of values, both precision and recall require condition 1 to be true. However, all they can assess is the resulting notion of effectiveness in that the values yielded by the scoring function are only meaningful for a single key. Thus, when we are evaluating the performance of a scoring function over a number of keys, it is usual to calculate the precision and recall measures individually for each key separately and then take the mean over all keys.

However, such an approach cannot, by definition, make comparisons across keys as required for Condition 3. Suppose that instead of using the values yielded by the scoring function we ranked all the documents with respect to a given key and gave each document an integer value to reflect its position in the list. In terms of Condition 1 this changes nothing but in terms of Condition 3 it does. Nevertheless, standard precision and recall measures would yield exactly the same result if scores are replaced with ranks. The fundamental idea behind the likelihood property is that the score is comparable not only between objects and the same key but also between different keys. We are interested in evaluating scoring functions with respect to Condition 3, in other words we wish to show that it is meaningful to rank across keys. Thus we construct two new effectiveness measures: *entire* precision and *entire* recall. Before defining them formally we first offer an example to illustrate the underlying concept. Let us suppose that a hypothetical (and somewhat contrived) IR system operated in batch model so that all queries to a collection of documents must be submitted in one go. This system then retrieves all documents which relate to any query in one single list. If a document is retrieved according to two queries it will appear twice in the list and so on. If we attempted to define the standard effectiveness measures over the entire batch operation the resultant measures would be entire precision and entire recall.

3.1 Definition of the Measures

Entire precision and recall are analogous to the standard precision and recall measures except that they apply over many keys simultaneously. For the set of objects and set of keys, we consider the cross product – the set of pairs containing each possible combination of object and key. We then use a *ranking strategy* to order this set of pairs based on the scoring function under consideration. Once the pairs are ranked, a cut-off point can be introduced to separate the set of pairs into retrieved and not retrieved. Each pair is said to match if and only if the key matches the object according to whatever matching criterion (e.g. relevance, culpability) is being used for normal retrieval.

We therefore define *entire precision* as the number matched pairs that were retrieved as a proportion total number of matched pairs. We define *entire recall* as the number of matched pairs that were retrieved as a proportion of the total number of retrieved pairs (whether matched or not).

3.2 Ranking Strategies

We propose two ranking strategies: one that takes the likelihood property into account and one that does not. We then attempt to show that the first strategy gives higher entire precision and entire recall measures than the second. If this is the case we have found evidence that the likelihood property exists. In the first strategy we use simply the value given by that scoring function for the object and key – this we term the *actual value* ranking. In terms of the hypothetical batch IR system mentioned above, the documents would be ranked by the absolute value of the scoring function and therefore it is possible that the documents ranked first, second and third with respect to one query will appear above the top ranked document yielded by a different query.

For the second strategy we rank all objects for each key first and then rank the pairs according to the initial ranking given. We will term this *ranked value* ranking. Using the example of the batch IR system again, here it would conduct retrieval for each query separately and then merge the lists so that all the top ranked documents for each query appears first, then the second ranked documents and so on.

Table 1 gives the raw data for a fictitious example where there are 4 objects and 2 keys. Table 2 shows how the pairs would be ranked by the two strategies. So, whereas pair (B, Y) is ranked fourth according to actual value scoring because it has the fourth highest score of all the pairs, it is ranked equal fifth for ranked value scoring since it is the third highest scoring pair for all those with key Y.

Values of Scoring function		Objects			
		A	B	C	D
Keys	X	**0.9**	0.02	0.03	0.05
	Y	0.34	**0.32**	0.33	0.01

Table 1: Scoring values for a fictitious example of a scoring function with 2 keys and 4 objects – bold indicates matching

Actual Value Scoring			Ranked Value Scoring		
Rank	Pair	Value	Rank	Pair	Value
1	**A, X**	**0.9**	1=	**A, X**	**0.9**
2	A, Y	0.34	1=	A, Y	0.34
3	C, Y	0.33	3=	D, X	0.05
4	**B, Y**	**0.32**	3=	C, Y	0.33
5	D, X	0.05	5=	C, X	0.03
6	C, X	0.03	5=	**B, Y**	**0.32**
7	B,X	0.02	7=	B, X	0.02
8	D,Y	0.01	7=	D, Y	0.01

Table 2: Comparison of Meta-scoring Functions showing each pair of object and key from the scoring values in Table 1 – bold indicates matching

If the underlying scoring function satisfies Condition 1 then, by definition, the matched pairs should tend to appear towards the top of the list. This is true for both strategies. However, if the scoring function also satisfies Condition 3 then the actual scoring value of a pair will be a better predictor of its being matched then the rank. Thus we expect matched pairs to be higher up the list for actual value scoring than ranked value scoring. The implication of this is that entire precision and entire recall will be higher especially when relatively few items are retrieved. In the example given, we can consider precision for the first 4 items. For entire actual value scoring, this is $\frac{2}{4} = 0.5$ whereas for ranked value scoring it is $\frac{1}{4} = 0.25$. For entire recall, the values are 1 and 0.5 respectively.

We now consider three case studies where entire precision and recall are used to determine the likelihood property.

4. Case 1: Information Retrieval

We consider here examples of IR models which are based on vectors of terms derived from a set of documents and the query. Such models are readily automatable and do not require human input such as the probabilistic models proposed by Fuhr (1989) and Turtle and Croft (1997). As we discussed in Section 2.2, these retrieval models are often divided into two categories: similarity models and probabilistic models. Similarity models produce a scoring function which measures degree of similarity between the vectors rather than probability of relevance and so the conditions described above are not appropriate and there can be no likelihood property. There are theoretical reasons to believe that one type of probabilistic model, namely language models, can under certain circumstances capture the likelihood property (Bache et al. 2007b). We provide evidence for this by comparing with a typical vector space model (TF-IDF), for which this property should be absent.

4. 1 Theoretical Basis for Models

Probabilistic models start from the premise that there is probability that a document is relevant to a query that can be expressed a function of the descriptions of the document and the query. Examples are OKAPI (Spark-Jones et al. 2000) and Lafferty and Zhai's justification for Language Modelling (2003). However, these models take as their only inputs the term vectors from the queries and documents. The formal derivation of these models assumes also the existence of certain quantities such as the probability that a document is relevant to an arbitrary query which cannot be estimated from the queries and documents alone. Thus the models are only made usable by eliminating these quantities by a series of order preserving transformations. In other words, inestimable quantities are removed from the model but the price paid is it yields only an ordinal function (Condition 1) rather than an estimate of actual probability (Condition 2). Also, for ease of calculation, logarithms are taken; this also preserves order.

Bache et al. (2007b) argue that under certain conditions the output of Language Modelling can be interpreted as a probability. This requires the assumption that the user formulated the query with the intention of finding a single ideal document which would satisfy the whole information need (which is almost certainly absent from the collection). We require also that we work in actual probabilities (not logarithms) and that the scoring function is normalised so that:

$$\sum_{o \in O'} r(o', k') = 1 \qquad (4)$$

This does not yield an actual probability of relevance satisfying Condition 2 but a measure of relevance likelihood satisfying Condition 3.

4.2 Data Analysed

Three TREC collections (TREC123, HARD03, HARD05) were analysed using both types of scoring functions. The Language Modelling scoring function chosen used Jelinek-Mercer smoothing (1980) with a smoothing parameter of 0.5. The default TF-IDF scoring function from The Lemur Toolkit (Zhai, 2007) was used as an example of a similarity function. We are not specifically interested here in comparing the respective performance of each model. It is well established that different models perform better or worse on different collections. Furthermore we note that we are dealing with three different collections where the entire precision and recall measures will vary across collections, whatever scoring function is used. Again this is not specifically of interest here. Instead, we wish to show the difference between the two scoring strategies for a given model and a given collection.

Calculations were performed using The Lemur Toolkit (2007) with the output files being post-processed by a purpose-built application to calculate the actual probabilities and also to calculate entire precision and recall at *n*.

4.3 Results and Discussion

Precision measures were calculated at various multiples of the number of queries. So for, say, 50 queries we would calculate recall at 50, 250, 500 etc corresponding to 1, 5 and 10 times the number of queries. This is because the ranked value scoring gives rise to ties and we want to make sure that any set tied pairs are either all retrieved or not retrieved. Tables 3, 4 and 5 give the results for each test collection. Note that since 1000 documents were retrieved for each query, both scoring methods will give the same result for precision at 1000 times number of queries. Figures 2 to 7 show the data in graphical form.

Pairs retrieved as multiple of number of queries	Language Modelling		TF IDF	
	Ranked	Actual	Ranked	Actual
1	0.0000	0.5600	0.4600	0.4200
5	0.0480	0.5160	0.3280	0.3520
10	0.0580	0.4720	0.3180	0.3360
15	0.0640	0.4507	0.3120	0.3253
20	0.0560	0.4190	0.3030	0.3440
30	0.0627	0.3873	0.2880	0.2980
100	0.0652	0.2858	0.2168	0.2052
200	0.0639	0.1989	0.1606	0.1507
500	0.0656	0.1194	0.0990	0.1032
1000	0.0686	0.0686	0.0622	0.0622

Table 3: Entire Precision for HARD 03

Pairs retrieved as multiple of number of queries	Language Modelling		TF IDF	
	Ranked	Actual	Ranked	Actual
1	0.1000	0.3600	0.5400	0.6200
5	0.0840	0.3120	0.4760	0.3840
10	0.0840	0.3080	0.3920	0.3760
15	0.0827	0.3013	0.3733	0.3627
20	0.0830	0.3090	0.3340	0.3550
30	0.0713	0.3093	0.3267	0.3340
100	0.0768	0.2424	0.2454	0.2524
200	0.0742	0.1801	0.1877	0.1780
500	0.0753	0.1108	0.1183	0.1023
1000	0.0761	0.0761	0.0735	0.0735

Table 4: Entire Precision for HARD 05

Pairs retrieved as multiple of number of queries	Language Modelling		TF IDF	
	Ranked	Actual	Ranked	Actual

1	0.0900	0.2450	0.1800	0.1650
5	0.0820	0.2600	0.1850	0.2240
10	0.0785	0.2710	0.1815	0.2355
15	0.0773	0.2703	0.1827	0.2413
20	0.0775	0.2653	0.1875	0.2458
30	0.0792	0.2488	0.1872	0.2333
100	0.0808	0.1913	0.1699	0.1818
200	0.0811	0.1493	0.1520	0.1377
500	0.0823	0.1078	0.1123	0.1031
1000	0.0828	0.0828	0.0820	0.0820

Table 5: Entire Precision for TREC123

We are specifically interested here in the difference between the two scoring strategies. It would be expected that a scoring function that adheres to Condition 3 to display higher score for the actual scoring compared to the ranked scoring. When plotted this would manifest itself as the gap between the two lines. The reasoning for this is that if the scoring function conforms to Condition 3, then documents will be more optimally ranked by the actual score in comparison to the ranked position by keys. This is because for some keys, there will be a higher proportion of relevant documents ranked higher with respect to the relevance score.

For Language Modelling, there is a marked improvement of actual value scoring over rank value scoring in all three collections. In other words, if we considered the top ranked documents for each query, the normalised scoring value would act as a predictor of relevance. The plots for all three collections provide supporting evidence that the normalised scoring function does give a measure of query likelihood as the *actual* scorings have a higher entire precision value compared to the *ranked* scorings. It is theoretically possible that the results are purely by chance, but given the number of documents retrieved (1000) and the number of queries (1000) and that the result occurred three times independently this is very unlikely.

This result is not repeated for TF-IDF where the different scoring procedures give results which are greater or less at various levels of entire precision and there is no great gap between them. The scoring function from TF-IDF is not based on a probability of relevance and thus should not exhibit the likelihood property. This leads to the question: could the likelihood property still be present if the two ranking strategies give broadly the same results? Theoretically this is possible but we can dismiss this possibility by a *reductio ad absurdum* argument. If this were to be true then we are asserting that all top ranked items will have broadly the same probability of relevance irrespective of the query. This will be true too for the second ranked, third ranked and so on. This would imply then that each query would have broadly the same number of relevant documents in the collection. A simple inspection of the data refutes this.

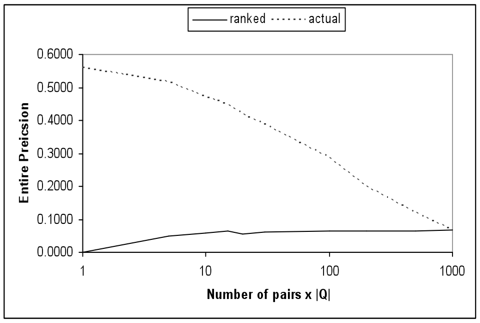

Figure 2: Entire Precision for HARD03 – Language Modelling

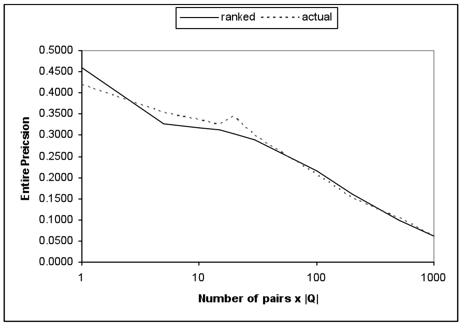

Figure 3: Entire Precision for HARD03 – TF-IDF

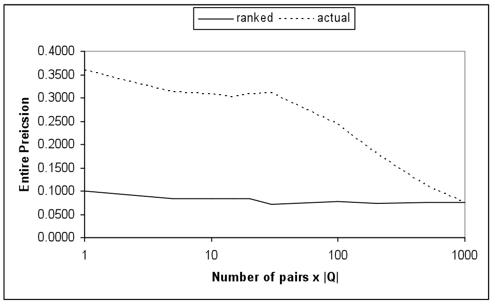

Figure 4: Entire Precision for HARD05 – Language Modelling

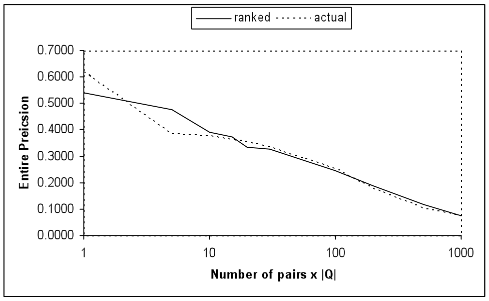

Figure 5: Entire Precision for HARD05 – TF-IDF

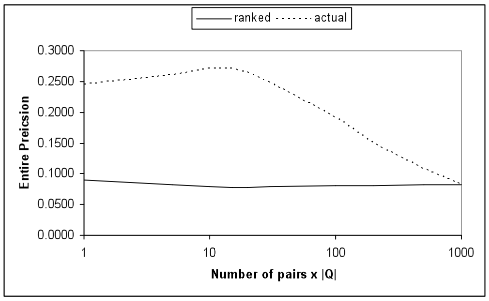

Figure 6: Entire Precision for TREC123 – Language Modelling

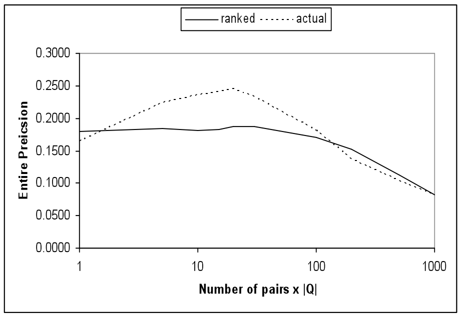

Figure 7: Entire Precision for TREC123– TF-IDF

5. Case 2: Text Analysis of Crime Data

It is well established that when offenders commit the same type of offence, they are likely to do so in a similar manner (Bennell & Canter, 2002; Canter and Fritzon, 1998; Yokota and Canter, 2004). Police can use this fact to prioritise suspects for an unsolved crime by comparing the way the crime was committed, known as Modus Operandi (MO) with previously solved crimes. However, given the enormous human effort required to look through and compare the descriptions of many previous offences, such a strategy is usually reserved only for the most serious offences such as murder and rape. Nevertheless the police do record both free text and structured data about volume crimes such as burglaries, vandalism and robbery as an electronic archive. These data contain information about the offenders' behaviour. The fact that free text is used in the descriptions means that techniques drawn from IR may be applied to find connections between a new unsolved offence and past solved ones. Here, we summarise the results and show how, firstly, this fits into our general model of retrieval and, secondly, that the suspect prioritisation system proposed also yields a measure of culprit likelihood.

5.1 Underlying Theory

Language Modelling was used (Bache et al. 2007a) to provide an automatic suspect prioritisation system based on comparing the textual description of an unsolved crime with descriptions of crimes of the same type. The rationale underlying the model is that features of the offender's behaviour will manifest themselves in the choice of words used by the police officer to describe the incident. Structured information in the form of a number of features which are recorded as either present or absent are mapped to extra tokens and added to the free text if present for that crime. We now consider how this problem differs from standard IR.

The set of objects to be retrieved here are not the unsolved crimes *per se* but sets of unsolved crimes linked to a particular offender who now becomes a suspect for the unsolved crime (the key). Where we know that a past crime was committed by more than one offender, that crime will appear in more than one set. So, to put this into our generalized version of Fuhr's framework, $k \in K$ is a text representation of the unsolved crime, and $k' \in K'$ is an indexed version of this. Also $o \in O$ is the set of text representations of crimes known to have been committed by one offender; $o' \in O'$ is the indexed version of the concatenation of these documents. The matching criterion is culpability. This is again stored in the police archive and is interpreted as a suspect being charged with a given crime. The scoring function produces a probability of each suspect being the culprit. Note that where there a crime has more than one offender it is assumed that they adopt the behavioural features of the dominant member of the group. In this case the model seeks to identify the ringleader.

The Language Modelling approach requires Bayes' theorem and thus a prior. In typical IR applications, it is usually assumed (somewhat unrealistically) to be the same for all documents; for an arbitrary query, all documents are equally likely to be retrieved. However, for this suspect prioritisation model, the prior is more meaningful and represents the probability that a given suspect will commit a crime. This can be estimated from the past frequency of offending.

The accounts of crimes entered by police officers are of variable quality. Some reveal clear features of the crime that indicate a particular pattern of behaviour. Others provide very little information. In this latter case the model will tend to give broadly similar values to each suspect but will still rank them even though this scoring does not tell us much. So, in addition to a ranked list, we would also want some measure of trustworthiness, particularly for the suspects ranked highest. Although the scoring function yields a probability and thus appears to satisfy Condition 2, an inspection of the results showed this not to be the case. For example, if we consider all the suspects given a probability of 0.99 or more, we would expect more than 99% to be culprits; the actual figure was nearer 60%. Nevertheless, it could be that a notional probability of 0.99

indicates a higher likelihood of being the culprit that 0.9 in which case Condition 3 would hold. So we use entire precision and recall to determine if this is the case.

5.2 Data Analysed

Eight crime sets were analysed reflecting different crime types. These were taken from a police archive of crimes collected over a 4-year period for an inner city district. Only solved crimes were considered. These were randomly allocated into a training set to act as the 'solved' crimes and a test set to act as the 'unsolved' crimes. The model was run 100 times to even out the effect of random allocation and the mean taken. Table 6 summarises the datasets.

Set No.	Crime Type	No. Crimes	No. Offenders	Crimes used for training model per offender
1	Theft from Vehicles	155	51	1
2	Other Theft	83	28	1
3	Shoplifting	803	294	2
4	Assault	436	205	1
5	Criminal Damage	255	82	1
6	Criminal Damage to Vehicles	37	17	1
7	Burglary	854	227	4
8	Robbery	138	62	1

Table 6: Summary of Crime Data Analysed

5.3 Results

Table 7 shows the average precision values for all 8 datasets. We note that the values for actual value scoring are always higher. If we apply a sign test to the results then we can conclude with greater than 99.5% confidence that the likelihood property is present.

Figure 7 shows the entire precision and recall graph for the criminal damage dataset which clearly shows how the two lines deviate at low levels of recall.

	Average Precision	
Set No.	Actual Value Scoring	Ranked Value Scoring
1	0.322	0.214
2	0.285	0.213
3	0.028	0.021
4	0.025	0.012
5	0.173	0.112
6	0.625	0.533
7	0.150	0.100
8	0.141	0.101

Table 7: Average Precision for Suspect Prioritisation based on text description

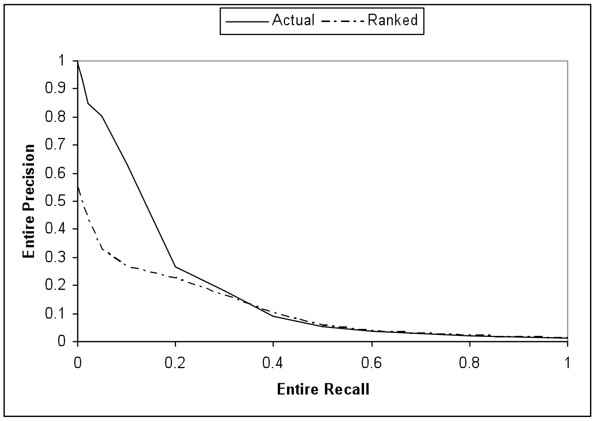

Figure 7: Entire precision and recall graph for Criminal Damage

6. Case 3: Suspect Prioritisation Based on Location

The final example considers the distance an offender travels to commit a crime and does not use text at all. Geographical information has been considered within the context of IR. Watters and Amoudi (2003) seek to extract words and phrases which indicate geographical location. The example we use here actually uses coordinates which were stored in the police archive along with the data used in case 2. Van Kreveld et al. (2005) consider the situation where a document is matched both according to its topicality and also distance and consider these to be two dimensions of matching which can be represented visually. Here we consider only the geographical dimension of the problem but accept that the analysis performed in this case and case 2 would need to be integrated at some stage. Nevertheless, concepts drawn from IR are useful here even though there is no text.

The assumption is that each offender travels from a base to commit a crime; this base will usually be the place of residence. The scoring function assigns probabilities to suspects according to the distance the suspect would have travelled. We consider a model which prioritises suspects based on the way in which the probability density of a crime decreases as the location moves further from the offender's base. Again we wish to determine if the scoring function has the likelihood property.

6.1 Underlying Theory

There has been extensive research into the distance offenders are prepared travel to commit a crime (Brantingham & Brantingham, 1981; Canter and Hammond, 2005; Turner, 1969). In particular, researchers have attempted to define a decay function which captures the decreasing relationship between distance travelled and frequency of offending. However, there is another reason why the probability of offending in a given location will decrease as we move further from the offender's base rooted in geometry. The further one is prepared to travel the more places there are to go. More formally, a circle drawn 2km from the base will have twice the circumference as a circle 1km. If the offender’s home base is known the distances can be calculated from each crime to that home base. Where the home base is not known, as in the data presented below, the centroid of previous crimes can be used. Thus to place this into the conceptual framework, $k \in K$ is the grid reference of the unsolved crime, and $k' \in K'$ is also the same so the mapping α_K is an identity operation. Also $o \in O$ is the set of grid references of crimes known to have been committed by an offender; $o' \in O'$ is the centroid of these points. Culpability is determined from the police archive as in case 2. The scoring function is derived from the decay function expressed as a probability density function and the application of Bayes’ Theorem.

We model the decay function with either a negative exponential or negative power function. Taking into account the geometry, it is possible construct a probability density function to estimate the probability an offender with a known base will offend at a given location. Using Bayes Theorem will then yield a scoring function for each base and crime scene. As with the suspect prioritisation model based on text, we could use a prior based on frequency of offending and this would make the model more accurate. However, here we will assume a uniform prior where all offenders are equally likely to offend since later we can compare this Bayesian model with a much simpler one. We again apply the test using both ranking strategies as in the previous two cases.

6.2 Data Analysed

The data was taken from the same source as in Section 5 although only a subset of the data sets was used. Each solved crime had a 6-figure grid reference, according to the

Ordnance Survey British national grid reference system, indicating its location to within 100 meters. The home location was not possible so instead we used the centroid of the known solved crimes to estimate it. Distance was calculated as the Euclidean distance but a value of 1 (i.e. 100m) was added so that locations in the same 100m square did not have a zero distance. Crimes were partitioned into a training set and test set as before and for each data set, the calculation of entire precision and recall was run 100 times and the mean taken.

6.3 Results

Figure 8 shows the precision and recall graph for the burglary data using the negative exponential model. The graph for the power model is almost coincident with the exponential model. Table 8 shows the average precision measures for both models using each ranking strategy. Both the negative exponential and power demonstrate similar performance and actual value ranking outperforms ranked value ranking. Using a sign test we can conclude that the likelihood property is present with 95% confidence. This is evidence that these Bayesian models do capture culprit likelihood.

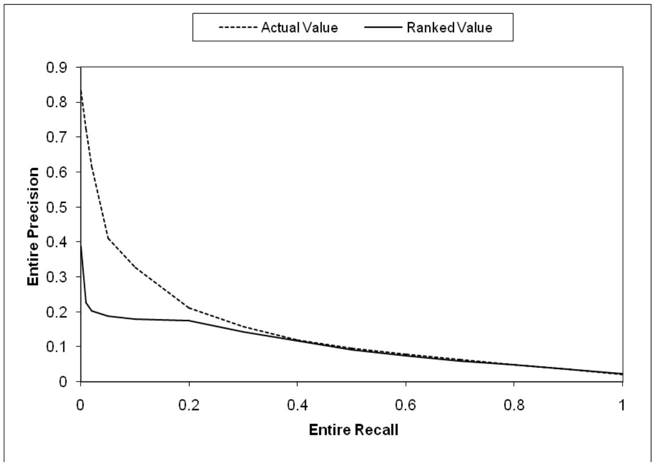

Figure 8: Entire Precision and Recall Graph for the Negative Exponential Model for Burglary

Crime Type	Neg. Exponential		Power	
Ranking Strategy	Actual	Ranked	Actual	Ranked
Theft from Vehicles	0.20	0.12	0.19	0.12
Criminal Damage	0.37	0.23	0.37	0.23
Damage to Vehicles	0.90	0.68	0.91	0.68
Burglary	0.14	0.10	0.13	0.10
Robbery	0.15	0.11	0.15	0.11

Table 8: Average Entire Precision for Location Models for Different Crime Sets

6.4 Corollary

We can consider a third suspect prioritisation model which is considerably simpler. This simple model merely ranks the offenders by distance and its scoring function assigns an integer to each offender where the nearest offender has the highest value. It is then easy to show that all three scoring functions are equivalent up to a strictly increasing transformation while there is a uniform prior. This explains why the ranked values in Table 8 are identical for both Bayesian models. Furthermore this simple model would yield the same entire precision statistics for ranked value ranking and incidentally this would, by definition, be identical to actual value ranking calculated for the simple model. The implication here is that, when using a uniform prior, the only advantage the Bayesian models offer over simple ranking by distance is that they indicate suspect likelihood. The ranking is identical. Of course, if priors were instead based on the frequency of past offending then the ranking provided by the Bayesian models would differ from the simple model (and possibly from each other).

7. Conclusions

By using the entire precision and recall measures it is possible to assess whether the likelihood property is present in a scoring function in a general retrieval situation. For IR, this means that we can infer that Language Modelling can yield a measure of likelihood provided that the scoring function is normalised. In the case of suspect prioritisation, which is essential a known item search, it means that the probabilities yielded by the model can be used as a measure of trustworthiness in the result for the highest ranked suspects even if the actual probabilities cannot be taken at face value.

Although the approach here was devised initially for the analysis of crime data, its application to other forms of retrieval, and, in particular, IR should be obvious. Models and scoring functions are being assessed all the time on different data sets to gauge the effectiveness of scoring, often using precision and recall. Entire precision and recall provide the researcher with new tools to determine whether or not the likelihood property is also present.

Acknowledgement: We would like to thank Prof. David Canter and Dr. Donna Youngs of the Centre for Investigative Psychology at the University of Liverpool for their assistance with the preparation of data and for the many interesting discussion which have been invaluable in the development of this research. We would also like to thank the anonymous reviewers whose constructive comments strengthened the paper.

References

Bache, R., Crestani, F., Canter, D., & Youngs, D. (2007) Application of Language Models to Suspect Prioritisation and Suspect Likelihood in Serial Crimes, International Workshop on Computer Forensics, Manchester.

Bache, R., Baillie, M. & Crestani, F. (2007) Language Models, Probability of Relevance and Relevance Likelihood, accepted for ACM Sixteenth Conference on Information and Knowledge Management, Lisbon, Portugal.

Carmel, D., Yom-Tov, E., Darlow, A. & Pelleg, D (2006) What Makes a Query Difficult?, Proceedings of the 29th annual international ACM SIGIR conference on Research and Development in Information Retrieval, Seattle, Washington, USA, pp390-397.

Bennell, C, & Canter, D. (2002) Linking commercial burglaries by modus operandi: tests using regression and ROC analysis, Science and Justice, Vol. 42 No.3.

Brantingham, P.J. & Brantingham P. (1981) Environmental Criminology, Waveland Press Inc., Prospect Heights, Illinois.

Canter, D. & Fritzon, K. (1998) Differentiating arsonists: A model of firesetting actions and characteristics, Legal and Criminal Psychology, Vol. 3, pp 73-96.

Canter, D. & Hammond, L (2006) A Comparison of the Efficacy of Different Decay Functions in Geographical Profiling for a Sample of US Serial Killers, Journal of Investigative Psychology and Offender Profiling, Vol. 3, No. 2.

Crestani, F., Lalmas, M., van Rijsbergen, C. J. & Campbell, I (1998), Is this Document Relevant … Probably: A survey of Probabilistic Models in Information Retrieval, ACM Computing Surveys, Vol. 30. No. 4.

Fuhr, N.(1989) Models for Retrieval with Probabilistic Indexing, Information Processing & Management, Vol. 25, No. 1 pp.55-72.

Fuhr N., (1992) Probabilistic Models in Information Retrieval, Computer Journal, Vol. 35, No. 3 pp243-254.

Jelinek, F. & Mercer, R. (1980) Interpolation estimation of Markov source parameters from sparse data, in Workshop on Pattern Recognition in Practice, Amsterdam, The Netherlands.

van Kreveld, M., Reinbacher, I., Arampatzis, A., & van Zwol R. (2005) Multi-Dimensional Scattered Ranking Methods for Geographic Information Retrieval, GeoInformatica, Vol. 9 No. 1, March.

Lafferty, J. & Zhai C (2003) Probabilistic Relevance Models Based on Document and Query Generation, in (ed. Croft W.B. and Lafferty J.), Language Modeling for Information Retrieval, Kluwer Academic Publishers, Dordrecht.

The Lemur Toolkit for Language Modeling and Information Retrieval (2007) [On-line]. Available: www.lemurproject.org

Ponte, J.M. & Croft, W.B. (1998), A Language Modeling Approach to Information Retrieval, in Proceedings of the Twenty First ACM-SIGIR, pp 275-281, Melbourne, Australia.

Sparck-Jones, K., Walker S. & and Robertson, S.E. (2000) A probabilistic model of information retrieval: development and comparative experiments. Information Processing and Management 36, Part 1 779-808.

Robertson, S.E. (1997) The Probability Scoring Principle in IR, in K. Spark-Jones, P. Willett (Eds), Readings In Information Retrieval, Morgan Kaufmann Publishers, San Francisco, California.

Turner, S (1969) Delinquency and Distance, in T. Sellen & M E Wolfgang (eds.), Delinquency Selected Studies, Columbia University Press, New York.

Turtle, H. & Croft, W.B. (1997) Inference Networks for Document Retrieval, in K. Spark-Jones, P. Willett (Eds), Readings In Information Retrieval, Morgan Kaufmann Publishers, San Francisco, California.

van Rijsbergen, C. J. (1979) Information Retrieval, Butterworths, London.

Voorhees, E. M. & Harman, D. K. (2005) TREC: Experiment and Evaluation in Information Retrieval, MIT Press, Cambridge, Massachusetts.

Watters C. & Amoudi G. (2003) GeoSearcher: Location-based ranking of search engine results, Journal of the American Society for Information Science and Technology, Vol. 54, No. 2 140-151, January.

Yokota, K. & Canter, D (2004) Burglars' Specialisation: Development of a Thematic Approach in Investigative Psychology, Behaviormetrika, Vol. 31, No. 2, pp153-167

Zhai, C. (2007) Notes of the Lemur TFIDF model, The Lemur Toolkit for Language Modeling and Information Retrieval [On-line]. Available: www.lemurproject.org/1.9/tfidf.ps